NUWC-NPT Technical Report 11,861
1 May 2008

Summary of a Modeling and Simulation Framework for High-Fidelity Weapon Models in Joint Semi-Automated Forces (JSAF) and Other Mission-Simulation Software

John R. Welch
Autonomous and Defensive Systems Department

NAVSEA
NAVAL SEA SYSTEMS COMMAND
Undersea Warfare Center

Naval Undersea Warfare Center
Newport, Rhode Island

Approved for public release; distribution is unlimited.

20080612 151

PREFACE

This study was prepared under NUWC Division Newport job order 798B240, principal investigator John R. Welch (Code 8212).

The technical reviewer was Gerald R. Martel (Code 8212).

The author thanks Duane M. Horton (Code 82), David S. Bellino (Code 82C), Christiane N. Duarte (8212), Gerald R. Martel (Code 8212), and Chi Ng (8223) for their support and assistance in this effort.

Reviewed and Approved: 1 May 2008

James S. Griffin
Head, Autonomous and Defensive Systems Department

REPORT DOCUMENTATION PAGE

Form Approved
OMB No. 0704-0188

The public reporting burden for this collection of information is estimated to average 1 hour per response, including the time for reviewing instructions, searching existing data sources, gathering and maintaining the data needed, and completing and reviewing the collection of information. Send comments regarding this burden estimate or any other aspect of this collection of information, including suggestions for reducing this burden, to Department of Defense, Washington Headquarters Services, Directorate for Information Operations and Reports (0704-0188), 1215 Jefferson Davis Highway, Suite 1204, Arlington, VA 22202-4302. Respondents should be aware that notwithstanding any other provision of law, no person shall be subject to any penalty for failing to comply with a collection of information if it does not display a currently valid OPM control number.
PLEASE DO NOT RETURN YOUR FORM TO THE ABOVE ADDRESS.

1. REPORT DATE	2. REPORT TYPE	3. DATES COVERED (*From – To*)
01-05-2008	Final	26 November 2007 to 30 May 2008

4. TITLE AND SUBTITLE

Summary of a Modeling and Simulation Framework for High-Fidelity Weapon Models in Joint Semi-Automated Forces (JSAF) and Other Mission-Simulation Software

5a. CONTRACT NUMBER

5b. GRANT NUMBER

5c. PROGRAM ELEMENT NUMBER

6. AUTHOR(S)

Welch, John R.

5.d PROJECT NUMBER
NUWC-NPT J.O. 798B240

5e. TASK NUMBER

5f. WORK UNIT NUMBER

7. PERFORMING ORGANIZATION NAME(S) AND ADDRESS(ES)

Naval Undersea Warfare Center Division
1176 Howell Street
Newport, RI 02841-1708

8. PERFORMING ORGANIZATION REPORT NUMBER

NUWC-NPT TR 11,861

9. SPONSORING/MONITORING AGENCY NAME(S) AND ADDRESS(ES)

Naval Undersea Warfare Center Division
1176 Howell Street
Newport, RI 02841-1708

10. SPONSORING/MONITOR'S ACRONYM
NUWC Division Newport

11. SPONSORING/MONITORING REPORT NUMBER

12. DISTRIBUTION/AVAILABILITY STATEMENT

Approved for public release; distribution is unlimited.

13. SUPPLEMENTARY NOTES

14. ABSTRACT

This report (1) summarizes a modeling and simulation (M&S) framework for testing high-fidelity weapon models in Joint Semi-Automated Forces (JSAF) and other mission-simulation software; (2) highlights the products and steps required to develop a weapon model and communicate with other weapon models in a mission-level simulation; (3) introduces the four configuration levels of the M&S framework; and (4) presents a cost-effective M&S laboratory design based on Mäk Technologies and MathWorks software.

15. SUBJECT TERMS

Undersea Warfare   Modeling and Simulation   Joint Semi-Automated Forces   Mission-Simulation Software

16. SECURITY CLASSIFICATION OF:			17. LIMITATION OF ABSTRACT	18. NUMBER OF PAGES	19a. NAME OF RESPONSIBLE PERSON
a. REPORT	b. ABSTRACT	c. THIS PAGE			John R. Welch
Unclassified	Unclassified	Unclassified	SAR	41	19b. TELEPHONE NUMBER (*Include area code*) 401-832-7973

TABLE OF CONTENTS

LIST OF ILLUSTRATIONS

Figure **Page**

LIST OF TABLES

LIST OF ABBREVIATIONS AND ACRONYMS

CPU	Central processing unit
DIS	Distributed interactive simulation
6-DOF	Six degrees of freedom
DSP	Digital signal processor
EEPROM	Electrically erasable programmable read-only memory
FOM	Federated Object Model
HLA	High-level architecture
ID	Identification
I/O	Input/output
IP	Internet Protocol
JSAF	Joint Semi-Automated Forces
LAN	Local area network
M&S	Modeling and simulation
NUWC	Naval Undersea Warfare Center
PC	Personal computer
PDU	Protocol data unit
RTI	Runtime infrastructure
RTW	Real-Time Workshop
SGI	Silicon Graphics, Inc.
TET	Total execution time
UDP	User datagram protocol
UUV	Unmanned undersea vehicle

SUMMARY OF A MODELING AND SIMULATION FRAMEWORK FOR TESTING HIGH-FIDELITY WEAPON MODELS IN JOINT SEMI-AUTOMATED FORCES (JSAF) AND OTHER MISSION-SIMULATION SOFTWARE

1. INTRODUCTION

Since 2006, the Naval Undersea Warfare Center (NUWC) Division, Newport, RI, has been internally funding an effort to establish a highly versatile modeling and simulation (M&S) framework that meets the requirements of the research and development community, the acquisition community, and the warfare analysis community.[1, 2, 3] Specifically, the goal of the M&S framework project is to develop the capability to test high-fidelity weapons models in Joint Semi-Automated Forces (JSAF)[4] software and other mission-simulation software.

This project will transpire in phases. The design, construction, and testing of the M&S framework were performed in phases 1 and 2. Phase 1 established the requirements of the M&S framework project. These requirements fit into two main categories: system level and mission level. At the system level, the framework must (1) support high-fidelity, physics-based simulations, (2) provide an ability to rapidly update, test, and evaluate new system models and processing algorithms, (3) communicate with legacy models written in Fortran and C, and (4) communicate with real hardware and proprietary software representations of real hardware.

At the mission level, the framework must provide (1) a means for interacting with other weapon systems in a simulated environment, (2) a method for defining a mission, and (3) a method for analyzing the results of the mission simulation against requirements. Appendix A provides a more extensive list of the M&S framework requirements.

Phase 2 of the M&S framework project looked at commercially available software that could meet the requirements of the M&S framework. Software products from MathWorks[5] and Advanced Dynamics International[6] were compared against the system-level M&S framework requirements. Software products from Mäk Technologies[7] and Engenuity Technologies (now Presagis[8]) were compared against the mission-level M&S framework requirements. The comparisons showed that any of the system-level software packages, when combined with any of the mission-level software packages, could achieve 88% of the stated requirements. When other factors were considered, however, the MathWorks and Mäk Technologies software packages were found to be the best choices for NUWC Division Newport's M&S framework.

MathWorks software is the software of choice for meeting the system-level M&S framework requirements for many reasons: (1) the level of technical support available for the MathWorks products is excellent (every technical question was answered, either by phone support or online documentation, within a reasonable amount of time); (2) the MathWorks products are easy to learn and understand; (3) the MathWorks products are based on MathWorks' Matlab, Simulink, and Real-Time Workshop software engines, which are already in wide use across NUWC Division Newport.

Mäk Technologies software is the software of choice for meeting the mission-level M&S framework requirements for several reasons, one of which is that Mäk Technologies offers the *HLA/DIS Toolbox*. This toolbox has various blocks that allow weapon models built in Simulink to communicate with other weapon models in a distributed simulation. The communication blocks use various standard protocols, including high-level architecture[9] (HLA) and distributed interactive simulation (DIS).[10] The toolbox seamlessly combines with the MathWorks products previously noted to make a complete M&S framework. The software package also provides an easy way to create and modify missions, create and destroy entities in the environment as the simulation progresses, and record data critical to the analysis of the mission.

This report (1) summarizes the features and capabilities of the M&S framework, (2) highlights the products and steps required to develop a weapon model and communicate with other weapon models in a mission-level simulation,* (3) introduces the four configuration levels of the M&S framework, and (4) presents a cost-effective M&S laboratory design based on Mäk Technologies and MathWorks software.

---

* The report assumes that the user has access to mission-simulation software such as VR-Forces[6] or JSAF.[4]

2. CONFIGURATION LEVEL 1

Level 1 is the fundamental configuration of the M&S framework: it is for users at the novice level with a minimal computation load requirement. Level 1 has the minimum software and hardware requirements for building a weapon model and connecting it to a mission simulation. The following sections describe the minimum hardware and software requirements for level 1, provide an example of a level 1 mission simulation, and discuss the advantages and disadvantages of a level 1 configuration.

2.1 HARDWARE AND SOFTWARE REQUIREMENTS

Configuration level 1 has the following hardware and software requirements:

1. Hardware Requirements - include one desktop or notebook personal computer (PC) with a built-in Ethernet adapter card and one fast Ethernet switch.

2. Software Requirements - include an operating system, MathWorks products, and Mäk Tecknologies products. A 32-bit operating system running Windows XP is recommended for running MathWorks and Mäk Technologies software products. Although MathWorks now supports 64-bit operating systems, only a 32-bit operating system has been used to test and evaluate the combined software packages that make up the M&S framework.

Building a weapon model as described in this document requires Matlab and Simulink.[5] Release 2007a and higher (2007b, etc.) of these products should be used to ensure compatibility with other software products of the M&S framework. These products allow users to build stand-alone weapon models that run non-real time. Non-real time in this case refers to Simulink interpreting the block diagram and executing the interpreted model.

Matlab and Simulink can be purchased from MathWorks for the approximate cost shown in table 1. The cost includes a 25% government discount and is relative to the year 2008 for a single user.[11]

Table 1. MathWorks Software Requirements for Configuration Level 1

Product	**Cost**	**Yearly Maintenance**
Matlab	$1425	$324
Simulink	$2250	$522
Total	$3675	$846

Connecting a weapon model to a mission simulation as described in this document requires software from Mäk Technologies. The HLA/DIS Toolbox from Mäk Technologies provides all the necessary blocks for connecting Simulink models to a mission simulation using standard DIS and HLA protocols. The toolbox loads into the Simulink Library as shown in figure 1. HLA and DIS communication blocks within the toolbox are clicked and dragged into the Simulink workspace.

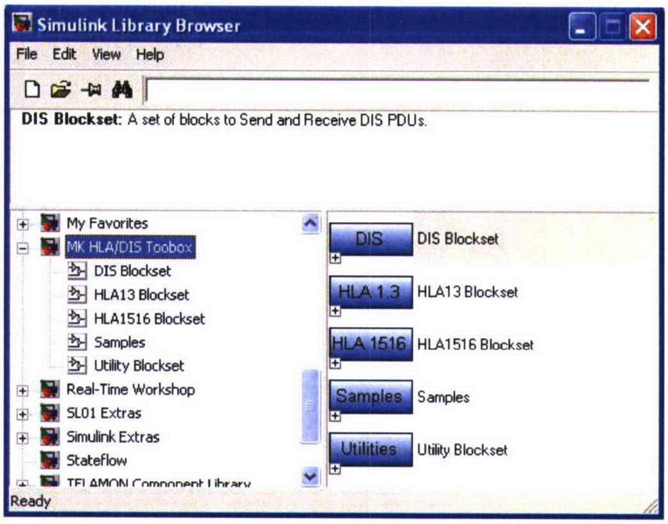

Figure 1. HLA/DIS Toolbox Loaded into the Simulink Library

The cost of the Mäk Technologies software depends on the number of users and the type of communication protocol used in the simulation. A single user who is connecting a Simulink model to a mission simulation with the HLA protocol requires the software listed in table 2. Additional users who are also connecting Simulink models to a mission simulation with the HLA protocol require only the software listed in table 3. A single user who is connecting a Simulink model to a mission simulation with the DIS protocol requires only the VR-Link Developer Toolkit. Additional users who are also connecting Simulink models to a mission simulation with the DIS protocol require only VR-Link Runtime. The cost of the Mäk Technologies software is relative to the year 2008.[12]

Table 2. Software Cost for a Single User Connecting a Simulink Model to a Mission Simulation Using the HLA Protocol

Product	Cost	Yearly Maintenance
VR-Link Developer Toolkit	$5000	$875
HLA/RTI License	$1000	$250
Total	$6000	$1125

Table 3. Software for Additional Users Connecting a Simulink Model to a Mission Simulation Using the HLA Protocol

Product	Cost	Yearly Maintenance
VR-Link Runtime	$1900	$475
HLA/RTI License	$1000	$250
Total	$2900	$725

The Mäk Technologies software has the following disadvantages:

1. The HLA/DIS Toolbox is not supported by the company because there is not a significant user base as of 2008.

2. The HLA/DIS Toolbox still comes as a part of the VR-Link Developer Toolkit, but there are no updates or technical support provided for the toolbox.[12]

3. The HLA/DIS Toolbox has limited protocol data unit (PDU) support. The toolbox can send and receive only entity state PDUs.

4. The HLA/DIS Toolbox lacks target filtering capability; it does not contain a specific block to filter out desired targets in a mission simulation.

Other disadvantages are discussed in later sections of this report.

NUWC Division Newport developed a temporary solution for target filtering during phase 2 of the M&S framework construction effort. Specifically, NUWC Division Newport combined several generic blocks within the HLA/DIS Toolbox to create a custom HLA Target Filter block and a custom DIS Target Filter block. The blocks can (1) filter out all air, surface, or subsurface objects; (2) filter out a particular platform such as a swimmer, submarine, or surface ship; and (3) provide the complete entity type for a particular platform in the simulation. The two filter blocks are shown in figure 2. Additional capabilities and operating instructions for the target filter blocks are given in appendix B.

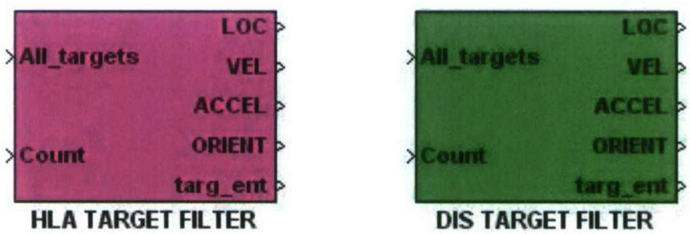

Figure 2. NUWC-Developed HLA and DIS Target Filter Blocks

2.2 EXAMPLE: LEVEL 1 CONFIGURATION

An example of a level 1 configuration consists of an unmanned undersea vehicle (UUV) model operating in a simulated mission. The Mäk Technologies VR-Forces software package runs the mission scenario.[7] VR-Forces is essentially a commercial version of JSAF. VR-Forces runs on a Windows platform and is both HLA- and DIS-compliant. The actual mission simulation consists of a UUV, submarine, human diver, and two aircraft carriers located in Newport, RI. A screen capture of the simulated mission is shown in figure 3.

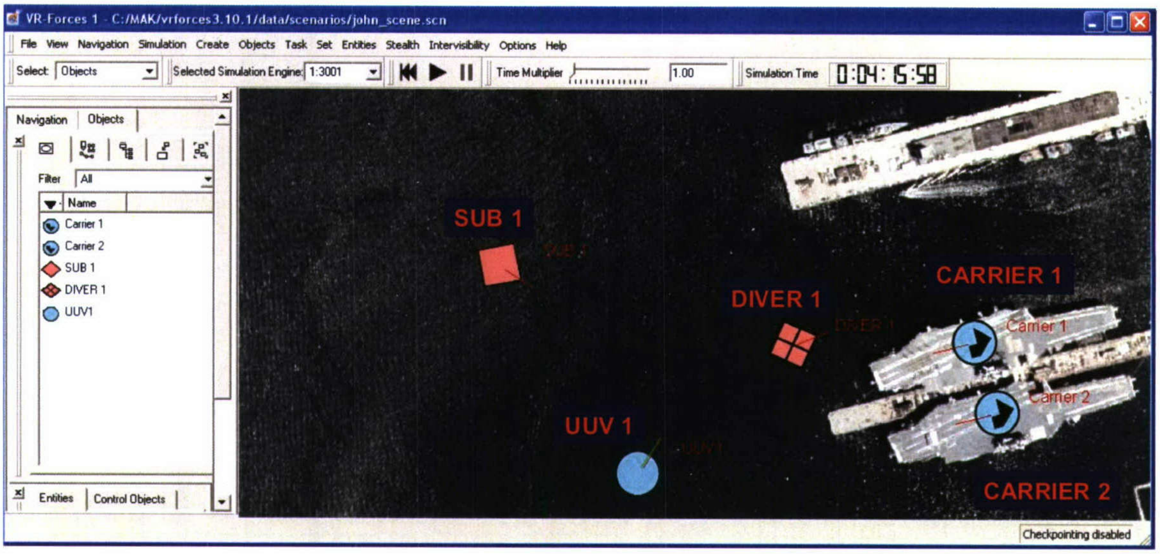

Figure 3. VR-Forces Running a Mission Scenario

The UUV in figure 3 is controlled by a high-fidelity Simulink model, which runs on a separate computer. The UUV model is physically connected to the VR-Forces simulation through a local area network (LAN). The physical connections between the UUV simulation and the mission simulation are shown in figure 4. The Simulink model uses the HLA 1.3 protocol to communicate with the mission simulation. Dynamic Internet Protocol (IP) addresses and a fast Ethernet switch are used to communicate over the NUWC Division Newport LAN. The NUWC LAN is needed to connect the Simulink model to VR-Forces (the Simulink model and the VR-Forces are located in different buildings).

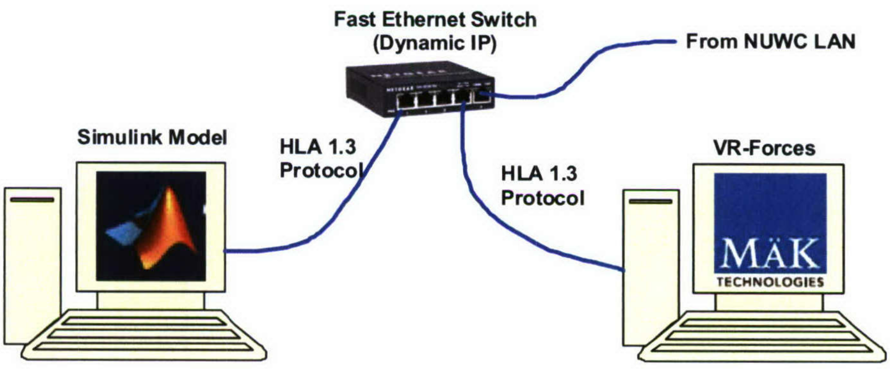

Figure 4. Physical Connections Between Simulink UUV Model and VR-Forces

The Simulink model consists of several blocks. The main blocks represent the UUV model, communication connections, an entity filter, and a timer function. The Simulink model is shown in figure 5. The UUV1 block is the UUV model. The UUV model inputs target location data, as well as course, depth, and speed commands, and outputs its entity state. The HLA Exercise Connection block establishes communications between VR-Forces and the Simulink model using the HLA 1.3 Protocol. The Entity Filter inputs the entity states of all objects in the mission and outputs only subsurface targets. The Publish UUV Entity block sends the UUV entity state PDU to the mission simulation. The Timer Function block slows the Simulink simulation time down to wall-clock time, which is when one simulation second equals one actual measured second.

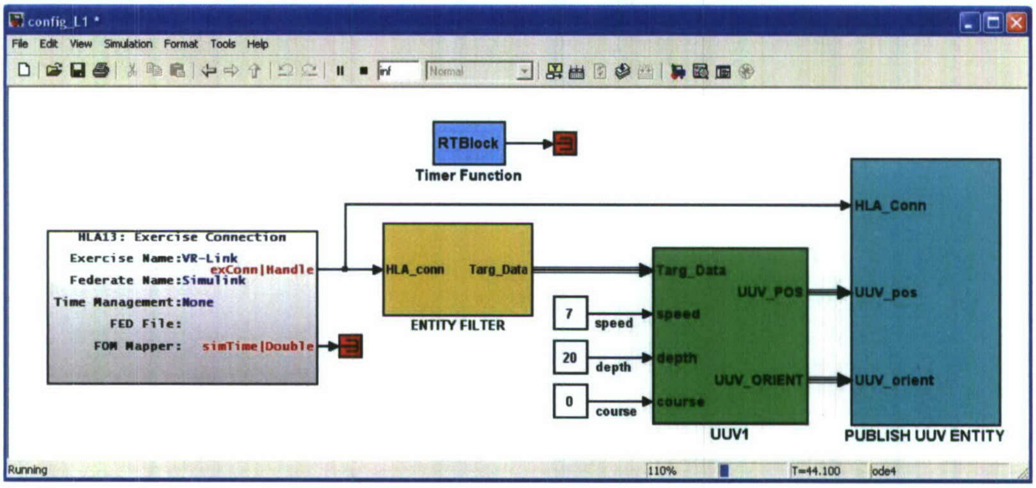

Figure 5. Simulink Model Connecting a UUV Model to VR-Forces

2.2.1 Timer Function Block

Mission simulations that have multiple weapon systems communicating using the HLA or DIS protocols usually maintain wall-clock time. Models running inside Simulink run as fast as Simulink can interpret and execute the code. This speed could be faster or slower than wall-clock time. Most high-fidelity models run more slowly than wall-clock time inside Simulink. Controlling the execution speed of a Simulink model that runs faster than wall-clock time is done with a timer function block. A timer function block, when added to the model workspace, reduces the execution speed of the Simulink model to wall-clock time. The Timer Function block and associated software written by Daga can be downloaded from the Matlab Central website.[13] An example of the Timer Function block is shown in figure 6. More information on and examples of specific uses of the Timer Function block can be found in NUWC-NPT TR 11,822.[14]

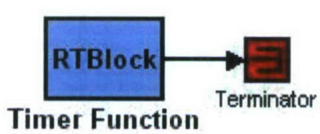

Figure 6. Timer Function Block Used To Reduce the Execution Speed of Simulink Models to Wall-Clock Time

2.2.2 *Entity Filter Block*

A view of the Entity Filter subsystem is given in figure 7. The Entity Filter consists of a Reflected Entity List block and an HLA Target Filter block. The Reflected Entity List block outputs a pointer to the entity states of all objects in the mission simulation. The HLA Target Filter block in this particular simulation outputs only subsurface objects. The entity types for the two subsurface objects in the mission simulation (excluding the UUV) are shown in the Display block. VR-Forces software identifies both the submarine and human diver as torpedoes.

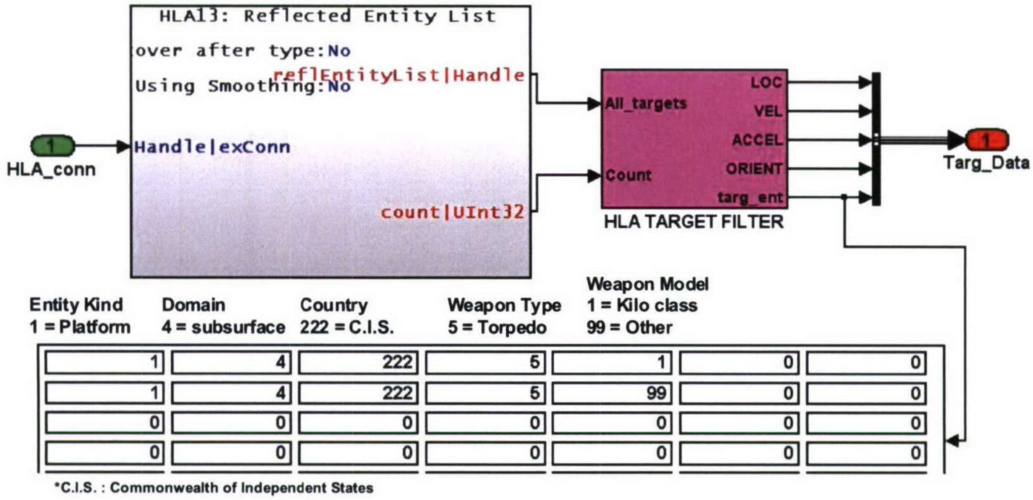

Figure 7. Entity Filter Subsystem Displaying the Filtered Objects in a Mission Simulation

2.2.3 *UUV Block*

Figure 8 is a view of the UUV1 subsystem, which contains several high-fidelity models: 6 degrees of freedom (6-DOF), sonar, energy, and tracker. Outputs from the 6-DOF model are used to update the location and orientation of the UUV in VR-Forces.

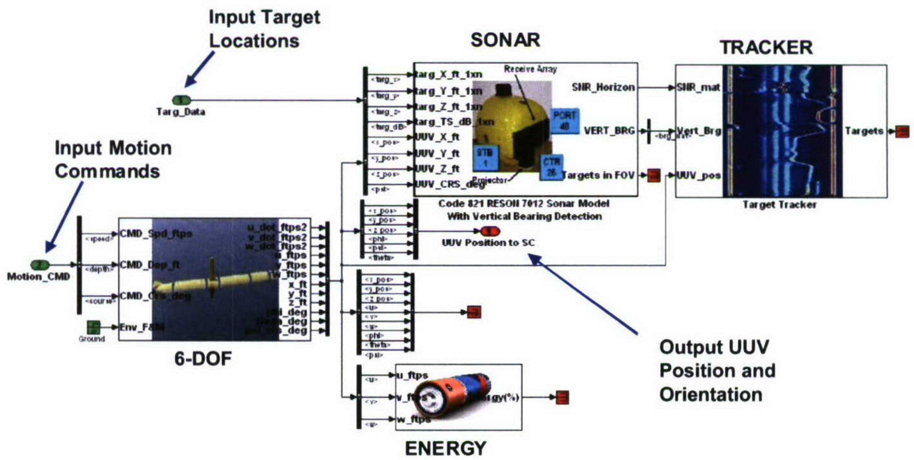

Figure 8. UUV1 Subsystem Model Used To Control the UUV in a Mission Simulation

2.2.4 Publish UUV Entity Block

Figure 9 is a view of the Publish UUV Entity subsystem, which contains several blocks: Coordinate Conversion, String, and Custom Entity Publisher. The Coordinate Conversion block converts topographic coordinates to geocentric coordinates. The String block specifies the name of the UUV. The Custom Entity Publisher block sends the UUV entity state PDU to VR-Forces using standard HLA 1.3 protocol. Figure 10 shows actual UUV identification (ID) and type information displayed inside the mission simulation.

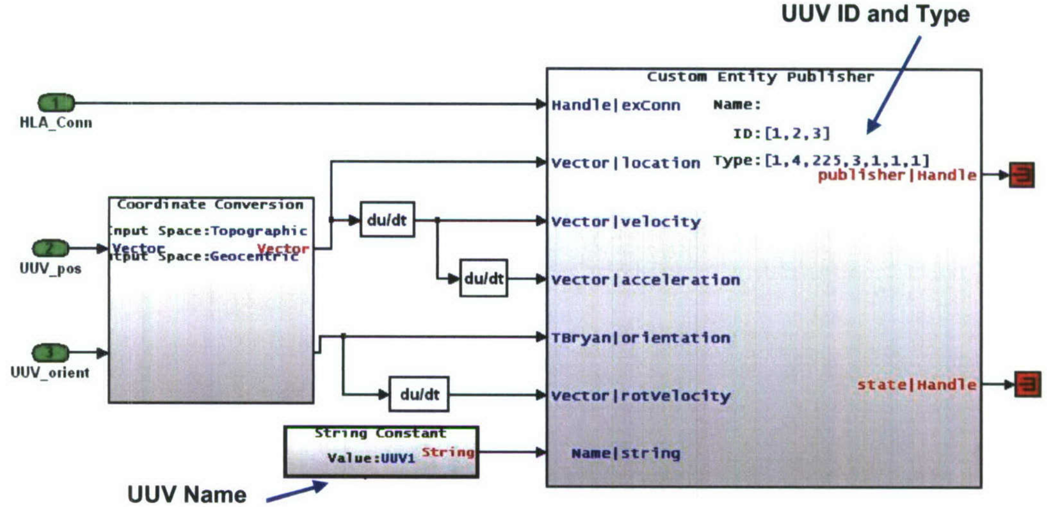

Figure 9. Publish UUV Entity Subsystem Used to Send UUV Entity State PDU to VR-Forces

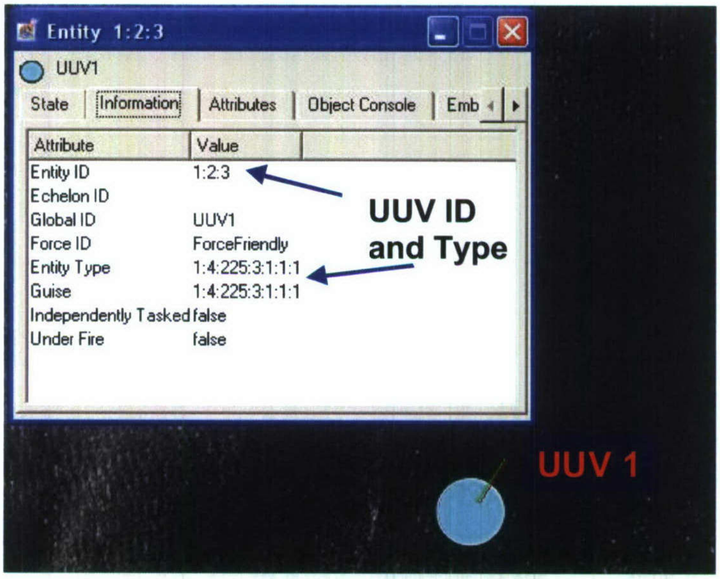

Figure 10. UUV Entity ID and Type Displayed Inside VR-Forces

2.3 LEVEL 1 ADVANTAGES AND DISADVANTAGES

The main advantages of configuration level 1 are cost and required skill level:

1. Level 1 requires the least amount of hardware and software to get started.

2. Level 1 requires the least amount of time and skill to connect a weapon model to a mission simulation.

The main disadvantages configuration level 1 are reduced computational loads and scalability:

1. Level 1 handles only small computational loads (low-fidelity models).

2. The level 1 process is not scalable; that is, the MathWorks individual license agreement allows only two separate processors, whether located on the same computer or different computers, to run simultaneous program sessions.[5]

3. CONFIGURATION LEVEL 2

Configuration level 2 of the M&S framework is required when the execution speed of the Simulink model is less than wall-clock time. Simulink models with execution speeds less than wall-clock time typically have high-fidelity weapon models and process a large number of entity state PDUs. In most cases, execution speed can be increased to wall-clock time by splitting the Simulink model in two and running each new model on a separate computer. One Simulink model contains the actual weapon model, and the other Simulink model contains all the blocks needed to send and receive entity state PDUs using the HLA or DIS protocol.

The splitting process generates a potential cost savings when multiple weapon models are being simulated. The Simulink model that sends and receives entity state PDUs becomes a Simulink gateway for all weapon models. Each weapon model sends its entity state to the Simulink gateway. The Simulink gateway sends all entity state PDUs to the mission simulation. A Simulink gateway requires the purchase of only one seat of the Mäk Technologies software listed in table 2. Without a Simulink gateway, each weapon model added to the mission simulation would require a seat of the Mäk Technologies software listed in table 3.[12] This splitting process does require additional hardware and software (see section 3.1).

3.1 HARDWARE AND SOFTWARE REQUIREMENTS

Configuration level 2 has the following hardware and software requirements:

1. Hardware Requirements - In addition to the hardware requirements listed for configuration level 1 (see page 3), configuration level 2 requires one additional Ethernet card for the desktop or notebook PC from configuration level 1, another desktop PC with a built-in Ethernet card, and another fast Ethernet switch.

The additional Ethernet card and fast Ethernet switch are recommended for communicating with computers on two different LANs. The weapon model and Simulink gateway communicate on a LAN with static IP addresses. The Simulink gateway and VR-Forces mission simulation communicate on a LAN with dynamic IP addresses. More information on the physical network connections is given in section 3.2.

2. Software Requirements - In addition to the software requirements listed for configuration level 1 (see page 3), configuration level 2 requires additional MathWorks products to allow the weapon model to communicate with the Simulink gateway. A weapon model communicates with the Simulink gateway using the User Datagram Protocol (UDP). MathWorks xPC Target software[5] provides UDP Send and UDP Receive blocks that can be dragged and dropped into the weapon model workspace from an xPC Target Simulink Library. The xPC Target Simulink Library is shown in figure 11.

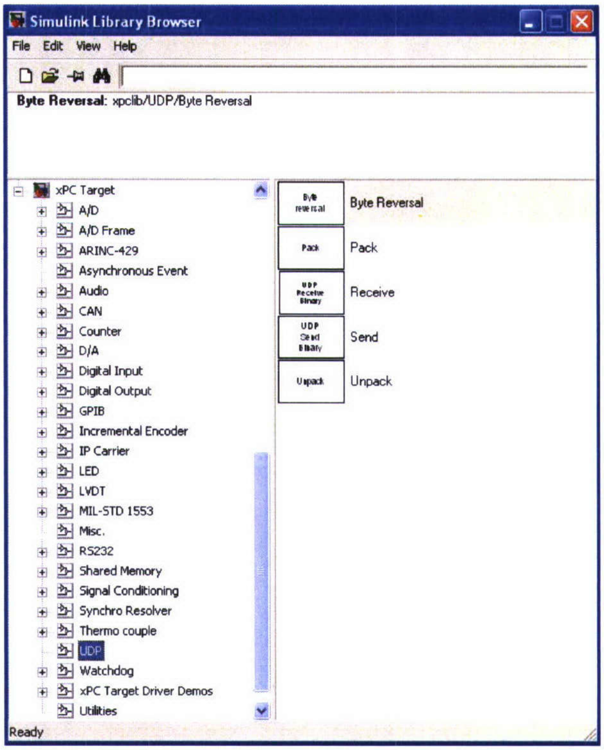

Figure 11. xPC Target Simulink Library

The xPC Target software cannot be purchased separately because it is designed to work with Real-Time Workshop (RTW).[5] The xPC Target software and RTW work together to generate, download, and run real-time code on a target PC. RTW must be purchased even though configuration level 2 does not generate and run real-time code. Real-time code is defined as "code that is already compiled and linked and executes without interpretation."

The xPC Target and RTW products can be purchased from MathWorks for the approximate cost shown in table 4. The cost includes a 25% government discount and is relative to the year 2008 for a single user.[11]

Table 4. MathWorks Software Requirements for Configuration Level 2

Product	**Cost**	**Yearly Maintenance**
Real-Time Workshop	$5625	$1305
xPC Target	$3000	$690
Total	$8625	$1995

3.2 EXAMPLE: LEVEL 2 CONFIGURATION

An example of a level 2 configuration consists of a UUV model operating in a simulated mission. The level 2 configuration uses the same UUV model and mission simulation as does the level 1 configuration (see section 2.2). The mission simulation consists of a UUV, submarine, human diver, and two aircraft carriers. A screen capture of the simulated mission is shown in figure 3.

The physical connections of the level 2 configuration are shown in figure 12. The Simulink weapon model and the Simulink gateway use UDP and static IP addresses to communicate. The Simulink gateway and VR-Forces use the HLA 1.3 protocol and dynamic IP addresses to communicate. Static IP addresses are preferred for the communication link between the weapon model and gateway because of the xPC Target UDP Send and UDP Receive blocks. The IP addresses of the sending and receiving computers are manually entered into these blocks. A fast Ethernet switch with dynamic IP is used to communicate over the NUWC Division Newport LAN, which is needed to connect the Simulink models to VR-Forces. The Simulink models and VR-Forces are located in different buildings. Note that the use of two fast Ethernet switches is preferred in this configuration but not required.

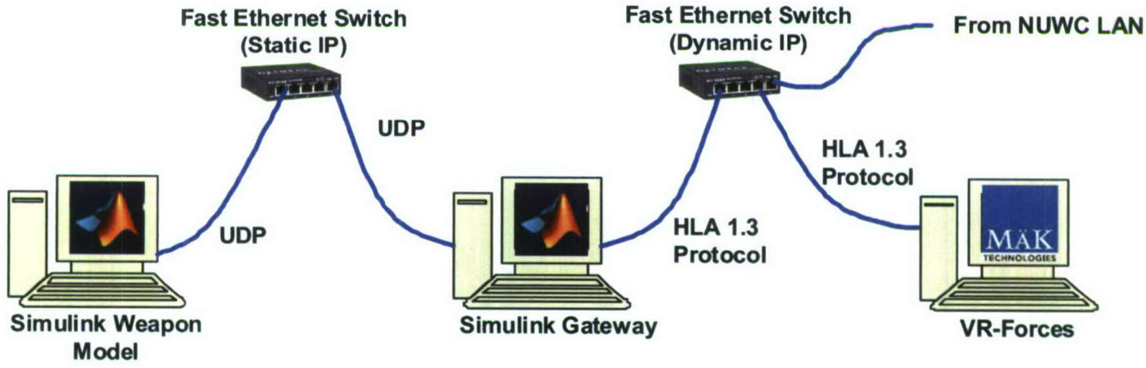

Figure 12. Physical Connections Between Simulink Weapon Model, Simulink Gateway, and VR-Forces

3.2.1 Simulink Gateway

The Simulink gateway is constructed from the Simulink model in figure 5 and is shown in figure 13. The Simulink gateway has all the blocks needed to send and receive entity state PDUs using the HLA or DIS protocol. The UUV1 model is removed and replaced with UDP Send and UDP Receive blocks. The Simulink gateway sends target data and motion commands to the UUV1 model using UDP Send blocks. The Simulink gateway receives UUV1 entity state information using UDP Receive blocks.

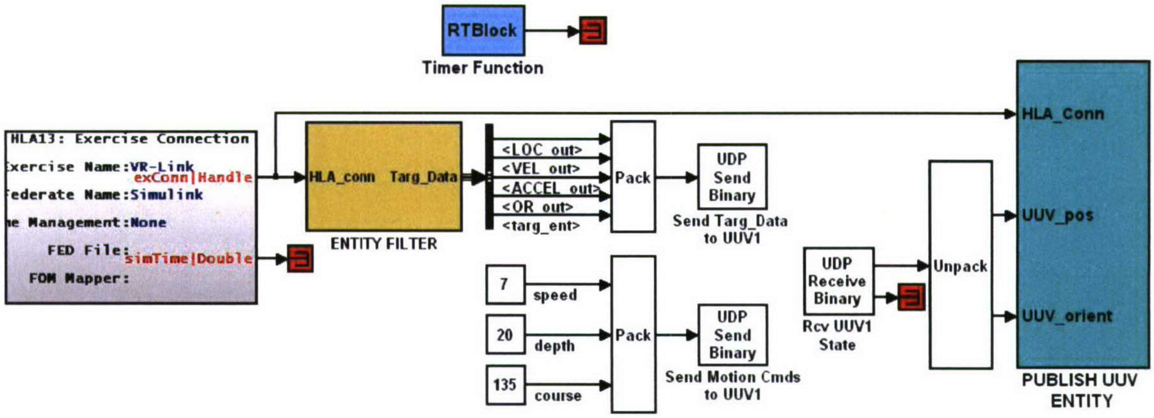

Figure 13. Simulink Gateway Communicating with One Weapon Model and VR-Forces

3.2.2 Simulink Weapon Model

The Simulink weapon model is constructed from the Simulink model in figure 5 and is shown in figure 14. The weapon model consists of the UUV1 model along with UDP Send and UDP Receive blocks. The weapon model receives target data and motion commands from the Simulink gateway using UDP Receive blocks. The weapon model sends UUV1 entity state information to the Simulink gateway using UDP Send blocks.

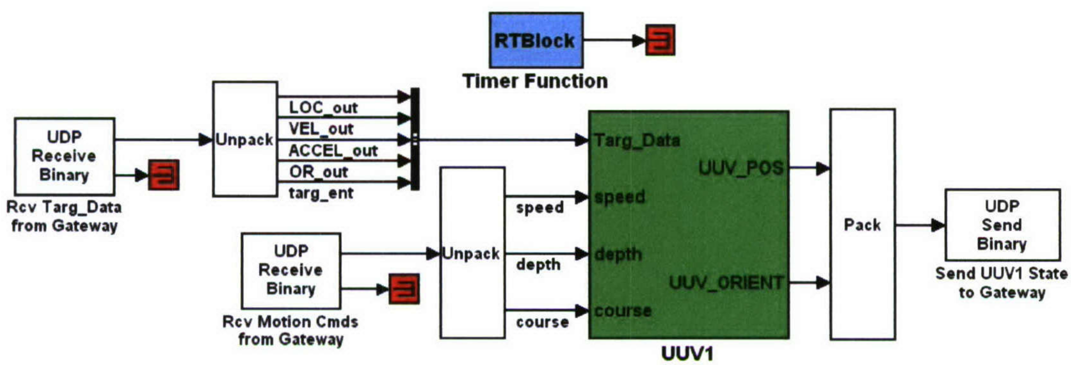

Figure 14. Weapon Model Configured for One UUV and Communicating with the Simulink Gateway

3.2.3 Gateway and Weapon Model for Multiple UUVs

Configuring the Simulink gateway and weapon model to handle multiple UUVs is a simple copy-and-paste routine. The gateway for handling two UUVs is shown in figure 15. The weapon model for handling two UUVs is shown in figure 16.

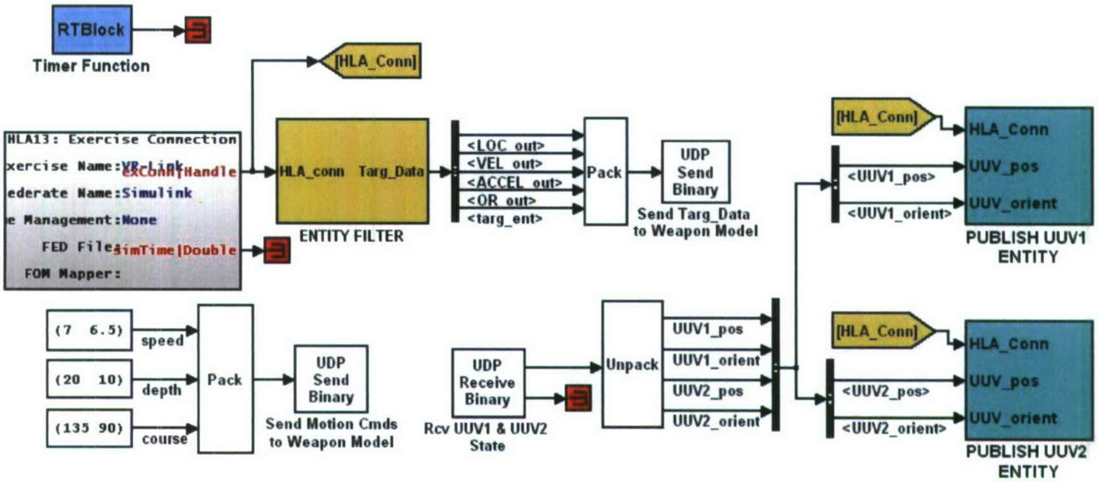

Figure 15. Gateway Configured To Handle Two UUVs

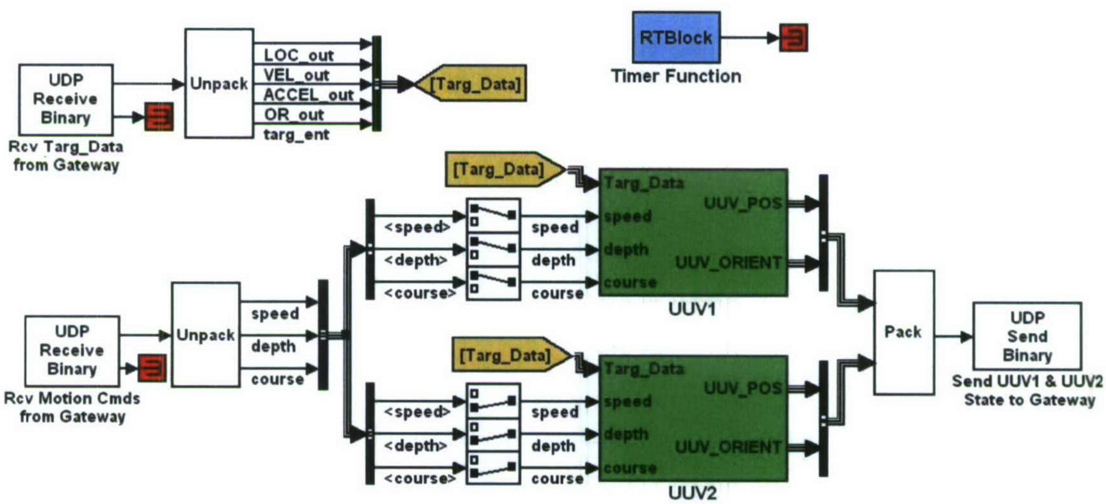

Figure 16. Weapon Model Configured To Handle Two UUVs

There is a limit to the number of UUVs that can be handled by the gateway and the weapon model. The limit is reached when either the gateway or the weapon model runs more slowly than wall-clock time. When the limit is reached on the gateway, the only option for increasing the execution speed to wall-clock time is to add another gateway and purchase the additional software from Mäk Technologies shown in table 3. When the limit is reached with the weapon model, the best option is to upgrade to configuration level 3.

3.3 LEVEL 2 ADVANTAGES AND DISADVANTAGES

The advantages of configuration level 2 are cost and model transparency:

1. Level 2 reduces software cost by using only one Mäk Technologies license (see table 3) to connect several weapon models to a mission simulation.

2. Level 2 makes it easy to debug the weapon model inside Simulink.

Running distributed, synchronized Simulink models inside Simulink has several disadvantages. It should be done only on a temporary basis to debug the weapon model. The weapon model should be converted to real-time code as described in section 4.

The disadvantages of running a distributed, synchronized model inside Simulink include:

1. Real-Time Workshop and xPC Target are significantly underutilized.

2. The level 2 process is not scalable; that is, the MathWorks individual license agreement allows only two separate processors, whether located on the same computer or different computers, to run simultaneous program sessions.[5]

3. The level 2 configuration exhibits a potential for the loss of synchronization. Specifically, observations show that long simulation times (5 to 30 minutes) can cause significant lag times in individual subsystems and the subsequent loss of synchronization. Lag time is suspected to be caused by a buildup of UDP messages.[15]

4. CONFIGURATION LEVEL 3

Configuration level 3 of the M&S framework is the recommended approach for increasing the execution speed of a weapon model. Configuration level 3 converts the Simulink weapon model to real-time code and executes it within a real-time kernel. The process is scalable and handles high computational loads while maintaining wall-clock time. The process cannot be applied to the Simulink gateway, which contains blocks from the HLA/DIS Toolbox. These blocks cannot go through RTW and cannot be downloaded and executed on the xPC Target real-time kernel.

Configuration level 3 uses a host machine and a target machine. The host machine runs the Simulink gateway, and the target machine runs the weapon model. The host machine also converts the Simulink weapon model into real-time code and downloads the code to the target machine. The host machine and target machine can be the same computers used in the level 2 configuration.

4.1 HARDWARE AND SOFTWARE REQUIREMENTS

Configuration level 3 has the following hardware and software requirements:

1. Hardware Requirements - In addition to the hardware requirements listed for configuration level 2 (see page 11), configuration level 3 requires one additional Ethernet card for the target machine. The xPC Target software requires a specific Ethernet card for the target machine. Many built-in Ethernet cards do not allow the host machine to communicate with the target machine. MathWorks provides a list of supported Ethernet cards for a target machine.[5] A recommended Ethernet card is the 3COM 3C905CTXM, which is inexpensive, reliable, and easy to obtain through many Internet vendors.[16]

2. Software Requirements - Software requirements for level 3 are those for level 2. Configuration level 3 also requires a compiler. The compiler has two purposes: (1) to convert Simulink block diagrams into C code, compile the code, and link the object code and (2) to provide a means for adding C and Fortran subroutines to a Simulink model. C and Fortran subroutines can be added to a Simulink model using a Simulink s-function. The steps required for creating an s-function for a C or Fortran subroutine are provided in the MathWorks documentation.[5] Although a C compiler does come with the purchase of the software listed in table 1, Microsoft Visual C[17] is recommended. MathWorks does not provide a Fortran compiler. A list of Fortran and C compilers that work well with Matlab and Simulink is shown in figure 17. Note that the Lcc compiler is the one provided with the MathWorks software. A list of available compilers is provided in Matlab by typing **mex –setup**.

```
Select a compiler:
[1] Compaq Visual Fortran 6.6 in c:\program files\microsoft visual studio
[2] Lcc-win32 C 2.4.1 in C:\PROGRA~1\MATLAB\R2007a\sys\lcc
[3] Microsoft Visual C++ .NET 2003 in C:\Program Files\Microsoft Visual Studio .NET 2003
[4] Microsoft Visual C++ 6.0 in C:\Program Files\Microsoft Visual Studio
```

Figure 17. Fortran and C Compilers That Work Well with Matlab and Simulink

4.2 HARDWARE AND SOFTWARE RECOMMENDATIONS

There is some recommended software for configuration level 3. The xPC Target Embedded Option[5] is recommended for weapon models that are debugged and ready for continuous use in a mission simulation. The xPC Target Embedded Option allows the user to deploy the weapon model as a stand-alone operation. Without the xPC Target Embedded Option, the executable code of the weapon model must be downloaded to the target PC from the host machine before the start of every simulation.

The xPC Target Embedded Option can be purchased from MathWorks for the approximate cost shown in table 5. The cost includes a 25% government discount and is relative to the year 2008 for a single user.[11] Note that one individual license is sufficient for a group of developers.

Table 5. Recommended MathWorks Software for Configuration Level 3

Product	Cost	Yearly Maintenance
xPC Target Embedded Option	$3000	$720
Total	$3000	$720

There is also recommended hardware for configuration level 3—mainly for deployed applications. The following hardware is recommended for configuration level 3:

1. a second hard drive or flash memory (thumb drive, electrically erasable programmable read-only memory (EEPROM)) for the target PC, and

2. a suitable target PC form factor.

A target PC uses the secondary hard drive or flash memory to automatically boot the xPC Target real-time kernel and run the deployed application. The primary hard drive of the desktop PC is reserved for the primary operating system (Windows, Linux).

A suitable target PC form factor for the application depends on many criteria, some of which are cost, space, environmental conditions, and input/output (I/O) requirements. Available form factors include the desktop PC, a rack-mount or industrial PC, a CompactPCI, and PC/104 and PC/104+. MathWorks[18] provides a summary of the advantages and disadvantages of each form factor as well as a list of vendors. The desktop PC form factor was used as the target PC for all configuration levels in this report mainly because (1) it was readily available, (2) physical space was not a constraint, and (3) the weapon model was not being permanently deployed.

4.3 EXAMPLE: LEVEL 3 CONFIGURATION

An example of a level 3 configuration consists of a UUV model operating in a simulated mission. The level 3 configuration uses the same Simulink gateway and weapon model shown in figures 13 and 14, respectively. The mission simulation consists of a UUV, submarine, human diver, and two aircraft carriers. A screen capture of the simulated mission is shown in figure 3.

The physical connections of the level 3 configuration are shown in figure 18. The host PC runs the Simulink gateway, and the target PC runs the xPC target weapon model. The xPC target weapon model and the Simulink gateway use UDP and static IP addresses to communicate. Static IP addresses are preferred for the communication link between the xPC target weapon model and the gateway because of the xPC target UDP send and UDP receive blocks. The use of dynamic IP addresses requires the user to re-enter the IP address in the UDP blocks every time it changes. The target PC uses a 3COM 3C905CTXM Ethernet card to communicate with the host PC. The Simulink gateway and VR-Forces use the HLA 1.3 Protocol and dynamic IP addresses to communicate. The fast Ethernet switch with dynamic IP is used to communicate over a NUWC Division Newport LAN, which is needed to connect the Simulink gateway to VR-Forces. The gateway and VR-Forces are located in different buildings.

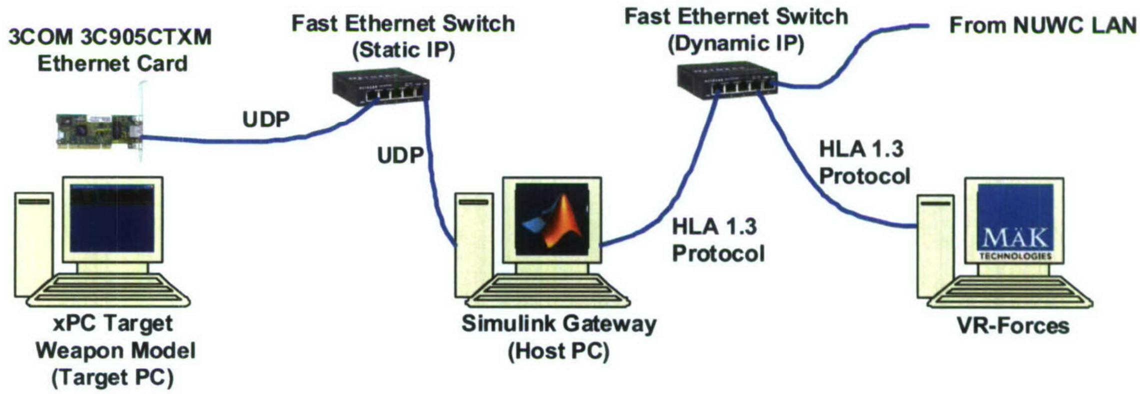

Figure 18. Physical Connections Between xPC Target Weapon Model, Simulink Gateway, and VR-Forces

The level 3 configuration example runs the same way as the level 2 configuration and produces the same results as those of the level 2 configuration example (section 3.2). The two configuration examples differ only in the way the weapon model executes. The configuration level 2 example executes the weapon model in non-real time inside Simulink; the configuration level 3 example runs the weapon model in real time inside the xPC target real-time kernel.

The process of building and running a real-time weapon model begins at configuration level 2. The weapon model is first tested in non-real time to ensure that its behavior is correct and that it communicates with the Simulink gateway. The test involves running the weapon model inside Simulink on the target PC and running the gateway inside Simulink on the host PC.

After a successful non-real-time test, the following procedure is used to build and run the real-time weapon model on the target PC:

1. Copy the Simulink weapon model onto the host PC.

2. On the host PC, go to the Matlab prompt and type **xpcexplr** . Then

    a. enter the communication settings for the target PC and

    b. make sure the IP address, Target Driver, and LAN subnet mask are correct (see figure 19).

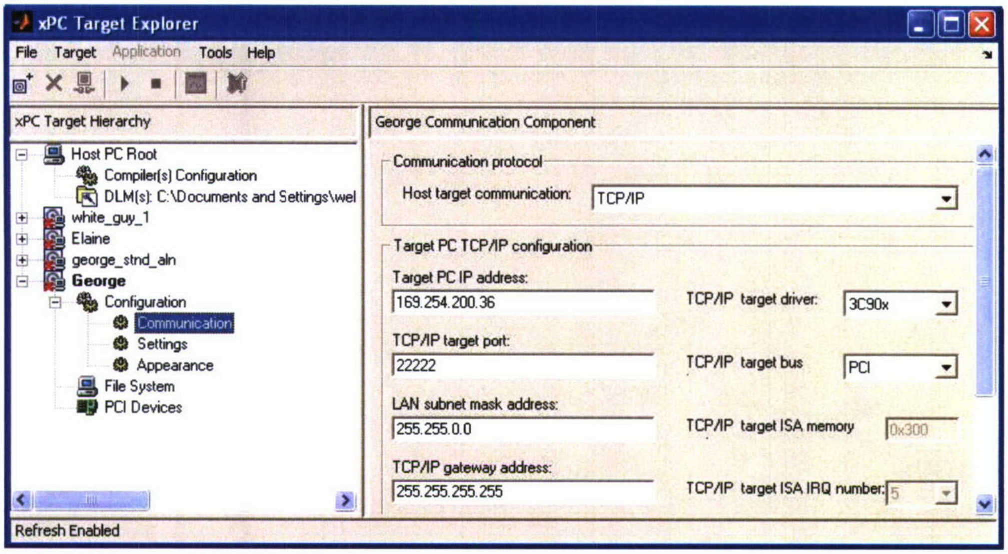

Figure 19. Example Communication Settings for a Target PC

3. Move up to the Target Configuration branch and create Bootdisk.

4. Reboot the Target PC with Bootdisk.

5. Go to the Host PC and open the Simulink weapon model:

    a. Remove the timer function block.

    b. Go to **Simulation, Configuration Parameters, Real-Time Workshop**.

    c. For the System Target File, choose **xpctarget.tlc**.

    d. Click on **Apply** and **OK**.

e. Run the weapon model for a few seconds to load any variables in the Matlab workspace.

f. Stop the weapon model and type **CTRL-B**.

6. On the host PC, go to the Matlab prompt and type **xpcexplr** .

   a. Go to the **Target** menu and select **Connect to Target**.

   b. Click on **Run** to start the weapon model on the target PC.

   c. Close **xPC Explorer** .

7. Go to the host PC and open the Simulink gateway inside Simulink.

8. Run the Simulink gateway inside Simulink.

More details on this procedure can be found in MathWorks documentation.[5]

4.4 LEVEL 3 ADVANTAGES AND DISADVANTAGES

The advantages of configuration level 3 are scalability, versatility, and computational loading:

1. The level 3 process is scalable. Up to 64 target PCs can be added to the simulation for each MathWorks individual license.

2. The level 3 process is versatile: it runs weapon models and Simulink gateways in non-real time for debugging; it runs weapon models on a target PC in real time to support high computational loads; and it uses an xPC embedded option to download a digital signal processor (DSP) signal processing algorithm onto a DSP chip, or PC/104 and runs in a real UUV or torpedo. Note that NUWC Division Newport's MARV UUV has a PC/104 on board.

3. The level 3 process handles high computational loads.

The main disadvantages of configuration level 3 include model transparency and communications:

1. Model transparency is one of the disadvantages of configuration level 3. Compared to running in non-real time, it is difficult to monitor the behavior of the weapon model once it is converted to real-time code and running on the target PC. There are, however, scopes and data files that can be used with the target PC to alleviate this issue. There are no known methods to determine why the target PC sometimes crashes.

2. Communication problems are another potential disadvantage of configuration level 3. A connection problem was observed between the target and host PCs when they were attempting to communicate over the NUWC Division Newport LAN. The connection problem disappeared when the host and target PCs communicated over a stand-alone LAN with static IP addresses, as shown in figure 18.[15]

Other observations indicate that only 5900 B/s can be sent to and from the target PC when it is in running real time.[19] When it is running in non-real time, the UDP Send and UDP Receive blocks can handle up to 100 MB/s.[15]

5. CONFIGURATION LEVEL 4

Configuration level 4 of the M&S framework is recommended for Monte Carlo simulations and for running real-time, distributed, synchronized weapon models. The level 4 configuration uses an external clock to control the execution speed of a target PC. The external clock can be used in Monte Carlo simulations to increase the execution speed of a target PC by orders of magnitude relative to wall-clock time. The external clock can also be used to synchronize a weapon model that consists of individual subsystems distributed across several target PCs.

The signal of the external clock can be generated in two ways: (1) by using a square-wave block inside a Simulink model on the host machine and (2) by using actual function generator hardware. Each method feeds the signal into the parallel port of the target PC. The first approach has not been tested. The second method is demonstrated in this report.

5.1 HARDWARE AND SOFTWARE REQUIREMENTS

The required hardware for configuration level 4 depends on the method used to generate the signal of the external clock. The method that uses a square-wave block inside a Simulink model on the host PC requires cabling from the host PC parallel port to the parallel ports of all target PCs in the simulation.

The method that generates a square wave by using actual function generator hardware requires a function generator and requires cabling from the function generator to the parallel ports of all target PCs used in the simulation.

5.2 EXAMPLE: LEVEL 4 CONFIGURATION

The example of a level 4 configuration is similar to that of the level 3 configuration; it uses the same weapon model shown in figure 14. The steps for testing the weapon model in non-real time, as well as loading and running a real-time weapon model on a target PC, are the same as those for configuration level 3. Configuration level 4 differs from level 3 only in the way it controls the execution speed of the weapon model.

Configuration level 4 controls the execution speed of the weapon model with an external clock. The use of an external clock must be specified before the weapon model is downloaded to the target PC. Specifying the use of an external clock is performed inside Simulink using the **Simulation** menu or by typing **CTRL-E**.

The following steps are used to specify the use of an external clock (these directions are valid only for MathWorks Release 2007a and higher).[20]

1. On the host PC, open the **Weapon Model** inside Simulink.

2. Remove the **Timer Function b**lock.

3. Go to the **Simulation** menu and then to **Configuration Parameters**.

4. Navigate to **xPC Target Options** under the **Real-Time Workshop Branch**.

5. Under **Execution options** (see figure 20):

    a. Select **Real Time** for the execution mode.

    b. Select **7** for the Real-time interrupt source.

    c. Select **Parallel_Port** for the I/O board generating the interrupt.

    d. Select **0x378** for the PCI slot/ISA base address.

6. Click on **Apply** and **OK.**

More details on the preceding procedure can be found in the MathWorks documentation.[5]

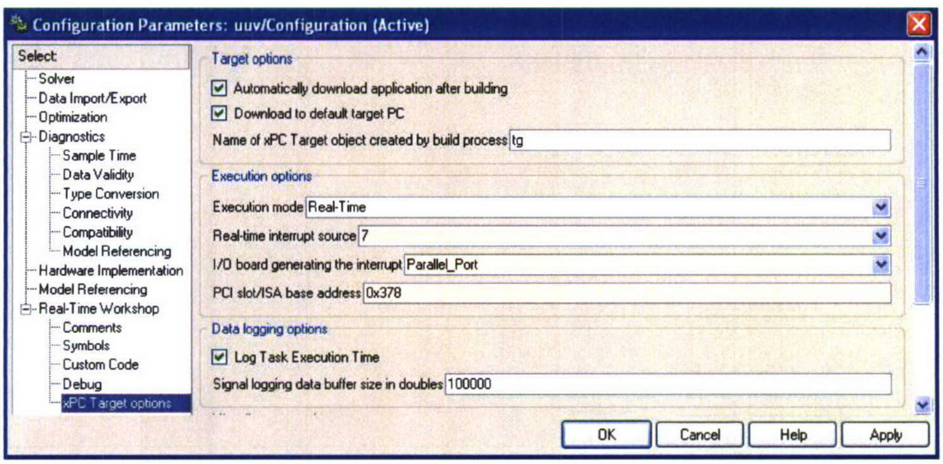

Figure 20. Execution Options for Controlling the Execution Speed of a Weapon Model with an External Clock

The physical connections of a weapon model controlled by an external clock are shown in figure 21. The weapon model runs in real time on the target PC. Execution speed is controlled by the signal received through the target PC parallel port. The positive and negative leads of the function generator connect to target PC parallel port pins 10 and 25, respectively. This setup supports both mission and Monte Carlo simulations.

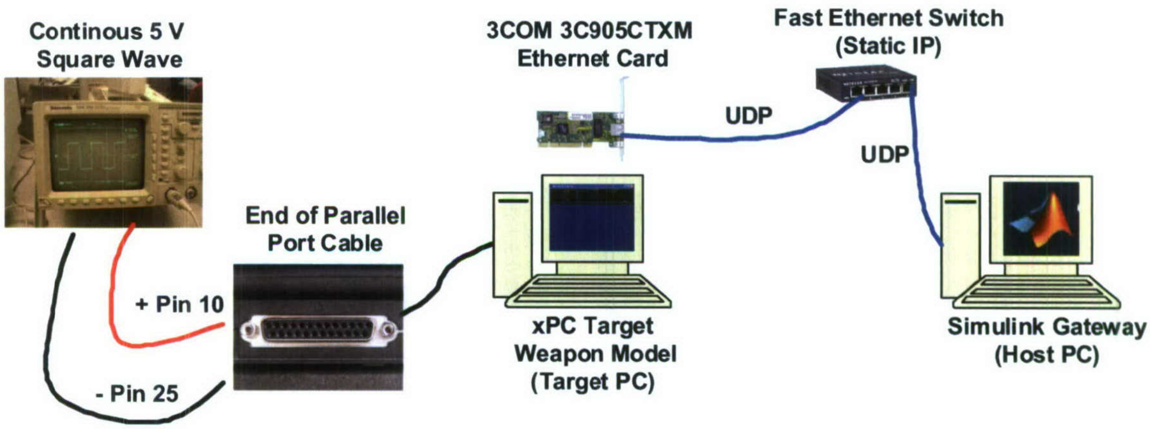

Figure 21. Physical Connections of a Weapon Model Controlled by an External Clock

The function generator outputs a continuous square wave. The amplitude of the square wave must be 5 volts. The cycle time of the square wave controls the execution speed of the weapon model. The amplitude and cycle time of a square wave are defined in figure 22. The weapon model maintains wall-clock time when the cycle time of the square wave is equal to the weapon model sample time. The weapon model runs faster than wall-clock time when the cycle time of the square wave is shorter than the weapon model sample time.

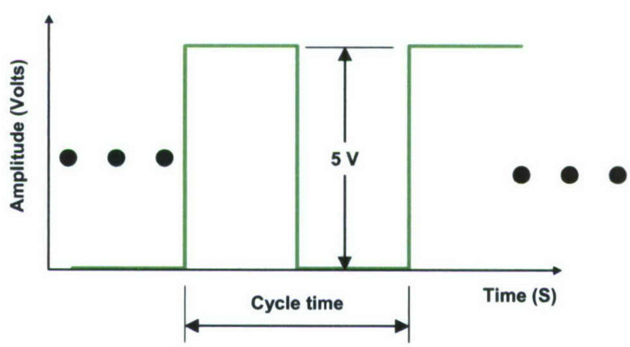

Figure 22. Amplitude and Cycle Time of an External Clock

There are limitations on the frequency of the square wave. The frequency of the square wave is measured in Hertz and is defined as 1/cycle time (second). Observations indicate that the frequency of the square wave should be less than or equal to the value given in equation (1).

$$f_{square\_wave} = \frac{1}{cycle\_time} \leq \frac{1}{2 * Average\_TET}, \tag{1}$$

where f_{square_wave} is the frequency of the square wave in Hertz, *cycle_time* is in seconds, and *Average_TET* (TET is total execution time) is the average amount of time (in seconds) required to advance the weapon model one sample step inside the xPC target real-time kernel. The average_TET is calculated and displayed on the target PC as shown in figure 23.

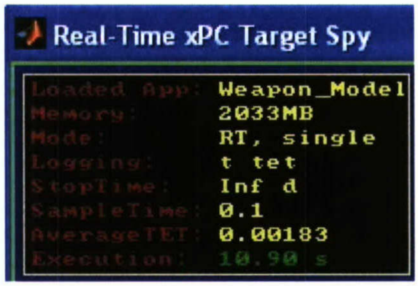

Figure 23. Weapon Model Sample Time and Average TET Displayed on the Target PC

Frequencies exceeding the observed limit in equation (1) often cause a central processing unit (CPU) overload and crash the target PC. For the case shown in figure 23, the frequency of the square wave should not exceed 273 Hz ($(2 \times 0.00183)^{-1}$).

5.3 LEVEL 4 ADVANTAGES AND DISADVANTAGES

Configuration level 4 has two main advantages:

1. Level 4 can support Monte Carlo simulations.

2. Level 4 can support real-time, distributed, synchronized simulations. An external clock can increase the execution speed of a target PC by orders of magnitude relative to wall-clock time. The external clock can also be used to synchronize the execution of a weapon model that consists of individual subsystems distributed across several target PCs.

The disadvantages of the level 4 configuration are the same as those listed for configuration level 3 (see section 4.4).

6. COST-EFFECTIVE M&S LABORATORY PLAN

This section describes the design of an initial M&S laboratory. The design has key components that serve as the building blocks of a fully expandable laboratory. The M&S laboratory plan is based on existing mission-simulation software, expected software demand at NUWC Division Newport, and available license options. The design is shown in figure 24.

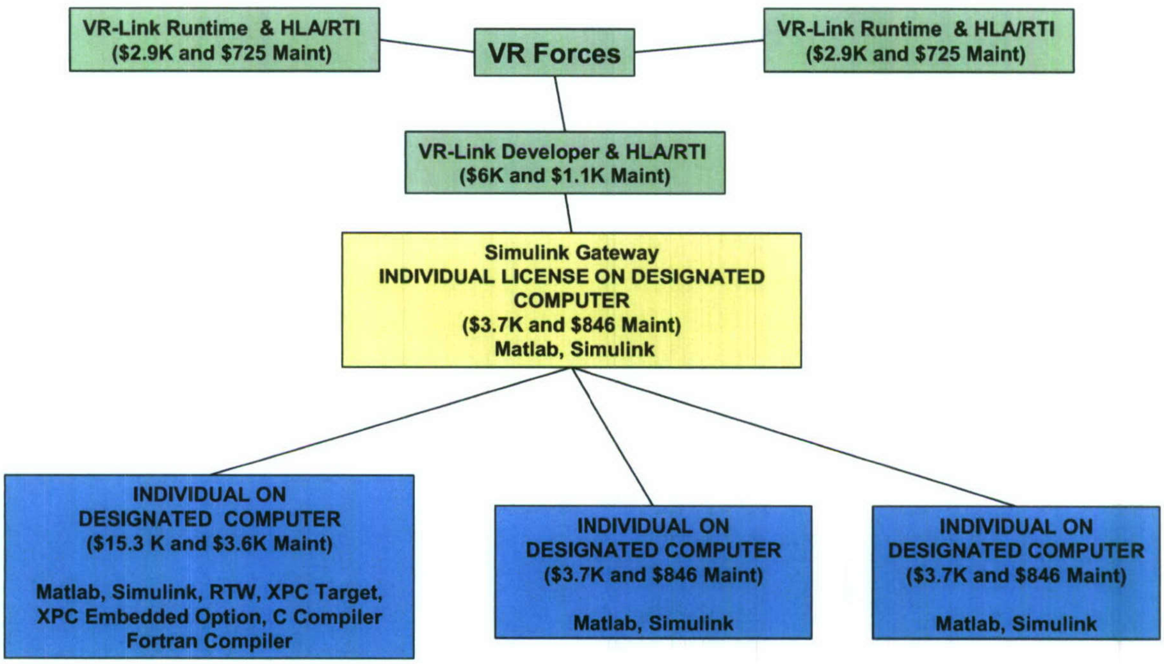

Figure 24. Design and Cost of an Initial M&S Laboratory

The mission-simulation software is the foundation of the M&S laboratory. Various software licenses are added to the foundation to expand functionality and meet expected software demands (see figure 24). The recommended mission-simulation software is Mäk Technologies VR-Forces. VR-Forces supports both HLA and DIS protocols. Any weapon model that plugs into VR-Forces is also capable of plugging into JSAF. One complete VR-Link Developer Toolkit and HLA/runtime infrastructure (RTI) license is recommended for connecting one weapon model to the mission simulation. At least two more VR-Link RTI licenses and two more HLA/RTI licenses are recommended for connecting at least two more weapon models to the mission simulation (see figure 24). The total cost of the recommended Mäk Technologies software is given in table 6.

Table 6. Total Software Cost for an M&S Laboratory

Company	Software Cost	Yearly Maintenance
Mäk Technologies	$11800	$2575
MathWorks	$26325	$6099
Total	$38125	$8674

The layout of the MathWorks software used in an M&S laboratory is driven by software demand, user skill level, and available licenses. At least two individual licenses of Matlab and Simulink are recommended for use inside the laboratory. The two individual licenses would be tied to a computer so that any two beginner-level users could access and use the software. A third individual license that includes Matlab, Simulink, Real-Time Workshop, xPC Target, xPC Target Embedded Option, a C compiler, and a Fortran compiler is also recommended. The third individual license would also be tied to a computer to allow any beginner user to build and run real-time weapon models. A fourth individual license of Matlab and Simulink is recommended for the Simulink gateway. This fourth individual license would also be tied to a computer. The total cost of the recommended MathWorks software is given in table 6.

A concurrent license should be considered as demand grows for the RTW and xPC Target products. As demand grows to 5 to 10 beginner users, the four individual licenses can be replaced with one concurrent license. A concurrent license allows 5 to 10 beginner users to access all MathWorks products including Matlab, Simulink, RTW, xPC Target, and xPC Target Embedded Option. The estimated cost of a single concurrent license is approximately $61K, with $14K in yearly maintenance fees.[21] The estimated cost includes a 25% government discount and is relative to the year 2008. The four individual licenses that are replaced by a concurrent license could be transferred to four skilled individual users.

7. SUMMARY AND CONCLUSIONS

This report presented four configuration levels of the M&S framework:

1. Configuration Level 1 - is for novice users and simple weapon models; it consists of one Simulink model comprising the weapon model and blocks for communicating with a mission scenario.

2. Configuration Level 2 - is required when the Simulink model of configuration level 1 runs more slowly than wall-clock time. Execution speed is increased by splitting the Simulink model in two and running each new model on a separate computer. One Simulink model called the weapon model contains the blocks that define the actual weapon. Another Simulink model called the Simulink gateway contains all the blocks for communicating with the mission simulation.

3. Configuration Level 3 - is required when the weapon model of configuration level 2 runs more slowly than wall-clock time. Configuration level 3 converts the Simulink weapon model into real-time code, downloads the code to a target PC, and runs the code in the xPC Target real-time kernel.

4. Configuration Level 4 - is required for Monte Carlo simulations and for running a real-time, distributed, synchronized weapon model; it uses an external clock to control the execution speed of a target PC. The external clock can be used in Monte Carlo simulations to increase the execution speed of a target PC by orders of magnitude relative to wall-clock time. The external clock can also be used to synchronize the real-time execution of a weapon model that consists of individual subsystems distributed across several target PCs.

Also presented in this report is the design of an initial M&S laboratory. The design has key components that can serve as the building blocks of a fully expandable laboratory. The M&S laboratory plan is based on existing mission-simulation software, expected software demand, and available license options.

This report suggests an improvement to the M&S framework could be made. The improvement is focused on converting the Simulink gateway into real-time code and running it in the xPC Target real-time kernel. A gateway running in real-time may allow more weapon models to connect to it. Fewer gateways require fewer VR-Link Runtime and HLA/DIS licenses, thus reducing software costs. The HLA and DIS blocks would have to be modified to allow them to go through RTW and download to a target PC.

REFERENCES

1. David Bellino, "Modeling and Simulation Framework for UDNS Concept Evaluation," quad chart for NUWC FY08 SI & CPP proposal, Naval Undersea Warfare Center Division, Newport, RI, 2008.

2. John R. Welch, "The Integration of a UUV Systems Level Engineering Model into a Theatre Environment Using an Open Architecture Integration Software and the DIS Protocol," NUWC New Professional Bid and Proposal, FY07, Naval Undersea Warfare Center Division, Newport, RI, 2008.

3. C. Agoros, "ADI Evaluation Task Statement" (unpublished), Naval Undersea Warfare Center Division, Newport, RI, May 2006.

4. "JSAF," http://www.jfcom.mil/about/fact_jsaf.html

5. "The MathWorks," http://www.mathworks.com

6. "Advanced Dynamics International," http://www.adi.com

7. " Mäk Technologies," http://www.mak.com

8. "Presagis," http://www.presagis.com

9. "HLA," Defense Modeling and Simulation Office, https://www.dmso.mil

10. "IEEE Standard for Distributed Interactive Simulation-Application Protocols," IEEE Std 1278.1a-1998, 19 March 1998.

11. T. Fotiades, "Quote for MathWorks Software Products," quote_1899908.pdf, MathWorks, Natick, MA, 9 November 2007.

12. John R. Welch, private communication (telephone conversation: Quote for Mäk Technologies Software), with Peter Swan, Mäk Technologies, 11 February 2008.

13. Leonard Daga, "Timer Function for Executing Simulink Models in Real-Time," Matlab Central, 2003, http://www.mathworks.com/matlabcentral/fileexchange/loadFile.do?objectId=3175&objectType=file

14. J. R. Welch, "Design and Implementation of a Simulink-Based Test Bed for Developing Prototype Algorithms for the Mid-Sized Autonomous Research Vehicle (MARV)," NUWC-NPT Technical Report 11,822, Naval Undersea Warfare Center Division, Newport, RI, 1 August 2007.

15. G. Martel, J. R. Welch, "Observations Regarding Communication Performance of UDP Blocks," Naval Undersea Warfare Center Division, Newport, RI, November 2007.

16. "3COM Corp 3C905CTXM Ethernet Card," http://www.zoomtek.com/product_info.php?products_id=1640

17. "Microsoft Visual C++.net Standard," http://www.amazon.com

18. "xPC Target Selecting Hardware Guide Version 1.1," http://www.mathworks.com

19. G. Martel, "Observations Regarding UDP Byte Size with xPC Target," Naval Undersea Warfare Center Division, Newport, RI, February 2008.

20. J. R. Welch, "Directions on How to Set Up an External Clock," private communication with Sam Mirsky at MathWorks, http://www.mathworks.com

21. T. Fotiades, "Quote for MathWorks Software Products," quote_1762599.pdf, MathWorks, Natick, MA, 25 June 2007.

APPENDIX A
M&S FRAMEWORK REQUIREMENTS

In 2006, an internally funded effort established the requirements of NUWC Division Newport's M&S framework. The requirements are listed in figure A-1; those that have been tested and are supported by the current M&S framework are indicated with check marks. Note that the requirement for handling different update rates has not been tested with the current M&S framework. Communication with a Silicon Graphics, Inc. (SGI) computer has not been tested either.

Requirements

- [x] Capable of building a fully integrated system from several subsystem models.
- [x] All subsystem models developed will abide by certain rules for integrating subsystems together in the open architecture environment.
- [x] The fully integrated model will be HLA 1516, DIS, or Federated Object Model (FOM) compatible so that it can be integrated into the JSAF environment.
- [x] Allow for the integration and alteration of various subsystems.
- [x] A method for testing and stimulating the fully integrated system in non-real-time to ensure proper operation
- [x] A method for testing and stimulating the fully integrated system in real-time to ensure proper operation in various environmental conditions.
- [ ] Handle the linking of subsystems with different update rates
- [x] Generates all necessary code for joining together and compiling subsystem models of different languages (Simulink, Fortran, and C)
- [x] Real-time system model code must be platform independent (capable of compiling on various operating systems)
- [x] The target operating system will not require the installation of MATLAB to compile the real-time system model code.
- [ ] Capable of communicating with an SGI machine
- [x] Options to run multiple iterations of a simulation, with the results dependent on one or more variables.
- [x] Store data results in a common location, and have user friendly methods to plot and analyze the data.
- [x] Real-time subsystem models can be swapped with the real hardware
- [x] Subsystem models can reside on different platforms and communicate via ethernet link in real-time
- [x] Capable of communicating with a subsystem without having access to source code

Figure A-1. NUWC's Requirements for the M&S Framework

APPENDIX B
HLA AND DIS TARGET FILTER BLOCKS

This appendix describes the general operation of a target filter block. The design and operation of the HLA and DIS Target Filter blocks are the same. Any modifications to the filter require disabling its link to the Simulink Library.

The HLA filter's main subsystem is shown in figure B-1. The subsystem uses a For Iterator block to repeat the subsystem process *N* times, where *N* is the number of received objects in the mission simulation. The Reflected Entity Selector outputs a pointer to the entity state of the N^{th} object in the mission simulation. The Reflected Entity block outputs the entity state of the N^{th} object in the mission simulation.

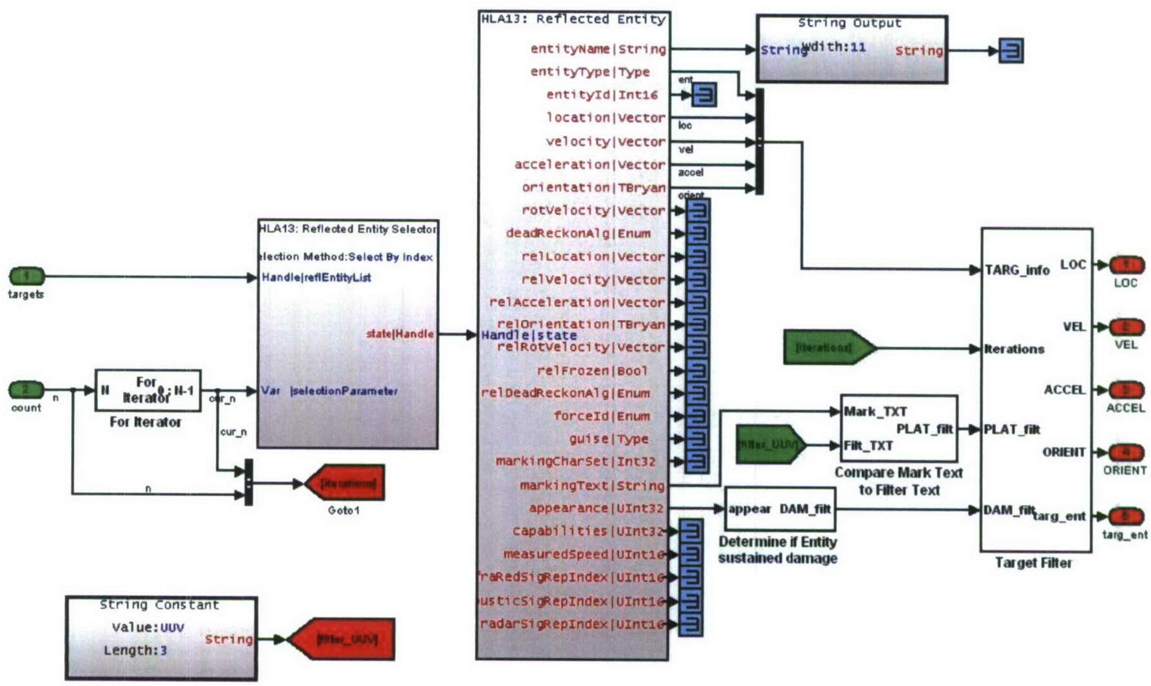

Figure B-1. Main Subsystem of an HLA Target Filter Block

The Target Filter block filters objects by name, damage, and entity type. The String Constant block and a Compare Mark Text (Compare Text) block are used to filter objects by name. The Compare Text block outputs the value "1" when the object marking text is the same as the string constant and outputs "0" when they are different. For the case in figure B-1, all objects named "UUV" are removed from the filter output.

A Determine If Entity Sustained Damage (Damage) block is used to filter out all damaged objects in the mission simulation. The Damage block receives a 32-bit number that represents the object's appearance. Bits 3 and 4 of the 32-bit appearance number indicate the degree of damage to the object. An object is completely damaged when both bits equal 1. The Damage block uses an AND operator to compare the 32-bit binary equivalent of the number 24 (...11000) to the 32-bit *appearance* number. The Damage block outputs the value 1 when the object is completely damaged and the value 0 for all other cases.

The Target Filter block performs the actual filtering. A portion of the code used inside the Target Filter block is shown in figure B-2. It uses the outputs from the Damage and Compare Text blocks to filter out completely damaged objects and objects named with a particular marking text. Entity type is also used to filter objects. The variable *DAM_filt* is the output from the Damage block. *PLAT_filt* is the output from the Compare Text block. The variable *ENT(2)* is the second value in the entity type vector (see figure 7). For this particular case, entity types that are surface (ENT(3)) and subsurface (ENT(4)) are not removed from the filter output.

```
if DAM_filt ~= 1 && PLAT_filt ~= 1 && count < TOT && (ENT(2) == 3 || ENT(2) == 4)
    LOC_data(count,:) = LOC';%[LOC' VEL' ACCEL'];
    VEL_data(count,:) = VEL';
    ACCEL_data(count,:) = ACCEL';
    OR_data(count,:) = ORIENT';
    entities(count,:) = ENT';
    count = count + 1;
    if count == TOT
        count = 1;
    end
end
```

Figure B-2. Code Used in the Target Filter Block To Filter Out Particular Objects in Mission Simulation

The count variable keeps track of the number of objects processed by the target filter. The Target Filter block outputs entity states for 30 objects. The count resets to 1 when the count reaches 30. When there are no objects, the target filter assigns the value 90000 to all 30 object location vectors (*LOC_data*). The value 90000 indicates an object that does not exist in the mission simulation.

INITIAL DISTRIBUTION LIST

Addressee	No. of Copies
Center for Naval Analyses	1
Defense Technical Information Center	2